Chapter 27
K-Optimal Chaos Ant Colony Algorithm and Its Application on Dynamic Route Guidance System

Hai Yang

Abstract Dynamic route guidance system is an important part of the intelligent transportation system; the core part of which is optimal path algorithm. This paper has analyzed the main influencing factors on the choice of optimal path, then provided an improved K-optimal chaos ant colony algorithm (K-CACA). The road impedance factor in K-CACA is based on the length, crowdedness, condition, and traffic load of the road sections. The optimizing procedure of the algorithm is speeded up by introducing the included angle threshold of direction. The chaos perturbation effectively refrains the algorithm from trapping into local optima. The results of simulation experiment show that K-CACA is effective and has much higher capacity of global optimization than Dijkstra algorithm and basic ant colony algorithm for optimal route choice.

27.1 Introduction

Dynamic route guidance system (DRGS) is an important part of the intelligent transportation system [1]. It offers the useful optimal route guidance information to the driver according to starting and destination point. The aim of DRGS is to achieve the goals of improving traffic system, voiding traffic jam, reducing the vehicles' travel time, and realizing the rational distribution of traffic flow on each road section by guiding the driver's travel decision [2]. So the core content of DRGS is the detection of optimal path in the traffic network.

With the constant enlargement of urban road network, the graph theoretic algorithms such as Dijkstra algorithm and mathematical programming methods are difficult to satisfy, the real-time requirements of DRGS. Initially proposed by

H. Yang (✉)
College of Information Science and Electricity Engineering, Shandong Jiaotong University, Jinan 250023, Shandong, China
e-mail: yanghai_sdjtu@163.com

W.E. Wong and T. Zhu (eds.), *Computer Engineering and Networking*, Lecture Notes in Electrical Engineering 277, DOI 10.1007/978-3-319-01766-2_27,

Marco Dorigo in 1991, the ant colony algorithm (ACO) was aiming to search for an optimal path in a graph based on the behavior of ants seeking a path between ant nest and a source of food. ACO has been successfully applied to solve different combinatorial optimization problems such as TSP [3] and also is suitable for DRGS because of its robustness, positive feedback, and distributed computing.

27.2 The Influencing Factors of the Optimal Path Selection

27.2.1 The Length of Road Section

The length of road section is the most common weight during the procedure of finding the optimal path because the length is easy to access and presents the travel cost directly [4]. But the optimal path does not only mean the shortest path in dynamic route guidance system. The travel time, crowdedness, conditions, and traffic loadings of the roads also affect the driver's decision of optimal path.

27.2.2 The Travel Time and Crowdedness

The travel time is more important than the length of road section in the real urban traffic network when drivers decide which path they should go. The travel time factor is dynamic according to the vehicle number per unit time. So we can compute the crowdedness of road by the travel time factor [5].

The crowdedness of road refers to the situation that the vehicles on the road cannot run at the normal speed so that both the travel time and parking delay become longer. The crowdedness factor can be expressed as $[T(i,j) - T_0(i,j)]/T_0(i, j)$, where $T(i,j)$ is the real travel time of road section (i,j) and $T_0(i,j)$ is the normal travel time of road section (i,j) without jam. In general, the smaller $T(i,j)$ is, the lower the crowdedness factor is, and vice versa.

27.2.3 Traffic Load

The traffic load refers to the number of vehicle staying on some road section for a moment. Particularly, the traffic load per kilometer presents the traffic density [6]. So the traffic load factor presents the change of traffic situation over time and its equation of state is as follows:

$$\frac{dx_{(i,j)}(t)}{dt} = I_{(i,j)}(t) - O_{(i,j)}(t) \tag{27.1}$$

where $x_{(i,j)}(t)$ is the vehicle number on road section (i,j) at time t. $I_{(i,j)}(t)$ is the number of vehicles coming into road section (i,j) per unit time. $O_{(i,j)}(t)$ is the number of vehicles leaving from road section (i,j) per unit time.

27.3 K-Optimal Chaos Ant Colony Algorithm

The urban traffic network is usually abstracted as a weighted, directed graph $G = (V,A,R)$, with node set V, road section set A, and $R = \{r_{ij}|(i,j) \in A\}$; r_{ij} denotes the road impedance factor, that is, heuristic information of road sections [7].

27.3.1 The Selection of Subsequent Node Based on the Direction

Because the dynamic route guidance system provides optimal path information according to the specific starting point and destination point, the procedure of finding optimal path is obviously directional [8]. When the ant selects next node, the probability of succeeding in finding optimal path is larger, if the ant's moving direction is close to the direction of destination point. Otherwise, the probability of succeeding in finding optimal path is lower, if the ant's moving direction is far away from the direction of destination point.

Let v_s denote the starting node with the coordinate (x_s, y_s); v_d denotes the destination node with the coordinate (x_d, y_d); v_i denotes the current node where the ant stays with the coordinate (x_i, y_i), and v_j denotes some node adjacent to v_i with the coordinate (x_j, y_j). Let $\overrightarrow{v_i v_d}$ denote the direction from current node v_i to destination node v_d, called the approximate destination direction. Thus, the included angle θ_{ij} of road section (v_i, v_j) and $\overrightarrow{v_i v_d}$ is as follows:

$$\theta_{ij} = \arctan \frac{k_d - k_{next}}{1 + k_d \cdot k_{next}} \tag{27.2}$$

where $k_d = \dfrac{y_d - y_i}{x_d - x_i}$ denotes the slope of the approximate destination direction $\overrightarrow{v_i v_d}$. $k_{next} = \dfrac{y_j - y_i}{x_j - x_i}$ denotes the slope of road section (v_i, v_j).

In order to enlarge the probability of succeeding in finding optimal path, we can set an included angle threshold ϕ . The subsequent nodes whose included angle is greater than the threshold ϕ will be omitted. The convergence speed of K-optimal chaos ant colony algorithm (K-CACA) will be accelerated by decreasing ϕ, and the solutions diversity will be enlarged by increasing ϕ.

27.3.2 The Heuristic Information of Road Section

In the new algorithm, the road impedance factor, which is gained by using weighted summation of the length, crowdedness, condition, and traffic load of the road sections [9], presents the heuristic information of road sections.

Let r_{ij} denote the road impedance factor of road section (v_i, v_j):

$$r_{ij} = \frac{\omega_1 \cdot \frac{1}{L_{ij}} + \omega_2 \cdot \frac{1}{C_{ij}} + \omega_3 \cdot Q_{ij} + \omega_4 \cdot S_{ij}}{\sum_{k=1}^{4} \omega_k} \tag{27.3}$$

where L_{ij} denotes the length of road section (v_i, v_j); C_{ij} denotes the crowdedness of (v_i, v_j); Q_{ij} denotes the condition of (v_i, v_j); S_{ij} denotes the traffic load of (v_i, v_j), and ω_k(k = 1, 2, 3, 4) is weight factor.

27.3.3 Pheromone Updating Strategy

The pheromone updating strategy of the new algorithm is as follow:

$$\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{l=1}^{L} \Delta\tau_{ij}^{l} \quad i = 1, \ldots, N \quad j = 1, \ldots, K \tag{27.4}$$

where ρ is the evaporation rate of pheromone.

27.3.4 Chaos Selection Strategy

When ant colony algorithm initializes, the strength of pheromones on every road section is equal so that the possibility of every road section is equal. It is difficult for ant colony to find an optimization path; besides, the convergence speed of algorithm is slow. So using chaos operator is necessary because it can increase the searching efficiency by the random and ergodic of chaos [10].

Take the logistics mapping, for example, iterative formula is as follows:

$$z_{i+1} = \mu \cdot z_i \cdot (1 - z_i), \quad i = 0, 1, 2, \cdots, \quad \mu \in (2, 4] \tag{27.5}$$

where μ is the control parameter, with the domain between (2, 4]. When μ = 4, $0 \le z_0 \le 1$, logistic function is the full mapping in (0, 1), which is at a totally chaos status. A chaos sequence will be created by the iteration and then be converted to a chaos ergodic parameter when solving optimization problems in space. After introducing chaos perturbation, the new status transition strategy becomes

$$p_{ij}^k(t) = \begin{cases} \dfrac{\left[\tau_{ij}^s(t)\right]^\alpha \cdot \left(r_{ij}^s\right)^\beta \cdot \left(1+Z_{ij}\right)^\gamma}{\sum\limits_{j\in allowed_k} \left[\tau_{ij}^s(t)\right]^\alpha \cdot \left(r_{ij}^s\right)^\beta \cdot \left(1+Z_{ij}\right)^\gamma}, j \in allowed_w \quad and \quad \theta_{ij} \le \phi \\ 0, otherwise \end{cases} \tag{27.6}$$

In the formula above, θ_{ij} is the included angle of road section (v_i, v_j) and the approximate destination direction $\overrightarrow{v_i v_d}$. ϕ is the included angle threshold.

27.3.5 K-Optimal Paths Guidance

The K-optimal chaos ant colony algorithm proposed in this paper provides drivers with K-optimal paths, which are almost exactly the same. Drivers decide which path they want to adopt by themselves, according to the real conditions of roads and the travel preference. The value of parameter K is referred to the scale of the traffic network, generally, K = 3. The newly improved algorithm can relax the conflict of traffic jam due to that many drivers choose the same optimal path.

27.3.6 The Realization Steps of K-CACA

The realization steps of K-CACA are proposed as follows:

Step 1: Set the iteration number n = 0, initialize the pheromone of all road section, and let K = 0.

Step 2: Put the starting node into $tabu_k(s)$, compute the road impedance factors of every road, and then choose one subsequent node j to move according to p_{ij}^k and the included angle threshold, finally put node j into $tabu_k(s)$.

Step 3: Update the pheromone of roads the ants passed by. Take record the current optimal path and let K = K + 1.

Step 4: If K equals to the specific value, then stop and go to step 6.

Step 5: Adjust the initial parameters according to the latest traffic situations, run this algorithm again in order to obtain new optimal paths. Go to step 2.

Step 6: Output K optimal paths.

27.4 Experimental Results

We successfully achieve this arithmetic and carry out simulation experiments using the road network of Jinan City shown in Fig. 27.1 from Google Earth. The topology of the road network shown in Fig. 27.1 is demonstrated in Fig. 27.2. The road impedance factors demonstrated in Table 27.1 are obtained by calculating the weighted summation of the length, crowdedness, condition, and traffic load of the road sections at 8:30 in the morning. Let m denote the number of ants and set m = 20, $\alpha = 1$, $\beta = 3$, and $\rho = 0.3$, and use logistic mapping. The aim of

Fig. 27.1 The road network of Jinan City

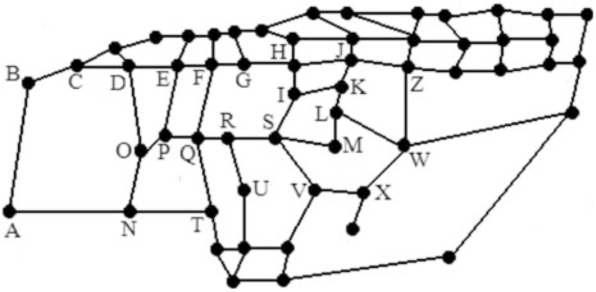

Fig. 27.2 The topology of road network of Jinan City

Table 27.1 The road impedance factor r_{ij} of road sections

Road section	r_{ij}	Road section	r_{ij}	Road section	r_{ij}	Road section	r_{ij}
(A, B)	4.96	(B, C)	2.16	(C, D)	2.02	(D, E)	3.13
(E, F)	2.87	(F, G)	3.01	(G, H)	3.08	(H, J)	3.97
(J, Z)	3.99	(Z, W)	4.01	(A, N)	4.86	(N, O)	3.22
(O, P)	2.13	(D, O)	3.99	(P, E)	3.91	(P, O)	2.71
(P, Q)	2.21	(E, P)	3.63	(Q, F)	3.70	(Q, R)	1.86
(N, T)	3.72	(T, Q)	3.91	(X, W)	3.77	(R, S)	3.64
(S, V)	3.71	(S, M)	3.57	(M, L)	2.28	(H, I)	1.13
(K, L)	1.97	(F, Q)	4.11	(L, W)	3.11	(V, X)	2.99

Table 27.2 The results of three algorithms

Algorithm	Optimal path	Distance (km)	Average arrival time
K-CACA	A-N-T-Q-R-S-M-L-W	13.67	22 min and 7 s
Dijkstra	A-N-O-P-Q-R-S-M-L-W	14.75	26 min and 3 s
Basic ACO	A-B-C-D-E-F-G-H-J-Z-W	14.14	23 min and 21 s

simulation experiment is to find the optimal path from node A to node W using K-CACA, Dijkstra, and basic ant colony algorithm (ACO) separately. The results are shown in Table 27.2.

The results of simulation experiment show that K-CACA is effective and has much higher capacity of global optimization than Dijkstra algorithm and basic ant colony algorithm for optimal path choice.

27.5 Conclusion

In this article, K-optimal chaos ant colony algorithm is proposed for finding optimal path in dynamic route guidance system. The road impedance factor is introduced into K-CACA based on the length, crowdedness, condition, and traffic load of the road sections. The included angle threshold both accelerates the convergence speed and enlarges the solutions diversity. The chaos perturbation effectively refrains the algorithm from trapping into local optima. The experiment results show K-CACA is much more suitable for DRGS.

Acknowledgements The research is supported by Chinese Natural Science Foundation (61103022) and Scientific Research Fund Project of Shandong Jiaotong University (Z201213).

References

1. Boyce, D. E., Ran, B., & Leblanc, L. J. (1995). Solving an instantaneous dynamic user optimal route choice model. *Transportation Science, 29*(2), 128–142.
2. Kuwahara, M., & Akamatsu, T. (1997). Dynamic user optimal assignment with physical queues for a many-to-many origin–destination pattern. *Transportation Research B, 31*(1), 1–10.
3. Dorigo, M. (1997). Ant colonies for the traveling salesman problem. *Biosystems, 43*(2), 73–81.
4. Ben-Akiva, M., Palama, A., & Kaysi, I. (1991). Dynamic network models and driver information systems. *Transportation Research A, 25*(5), 251–266.
5. Deek, H. L., & Kanafani, A. (1993). Modeling the benefits of advanced traveler information systems in corridors with incidents. *Transportation Research C, 1*(4), 303–324.
6. Daganzo, C. F. (1998). Queue spillovers in transportation networks with a route choice. *Transportation Science, 32*(1), 3–11.
7. Kim, J., Mitchell, J. S. B., & Polishchuk, V. (2012). Routing multi-class traffic flows in the plane. *Computational Geometry-Theory and Applications, 45*(3), 99–114.

8. Cordeau, J. F., & Maischberger, M. (2012). A parallel iterated tabu search heuristic for vehicle routing problems. *Computers and Operations Research, 39*(9), 2033–2050.
9. Blum, C., & Dorigo, M. (2004). The hyper-cube framework for ant colony optimization. *IEEE Transactions on Systems, Man and Cybernetics B, 34*(2), 1161–1172.
10. Yunwu, W. (2009). Application of chaos ant colony algorithm in web service composition based on QoS. *International Forum on Information Technology and Applications, 172*(2), 225–227.